Science of Computer Programming 91 (2014) 3–44

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Formal patterns for multirate distributed real-time systems

Kyungmin Bae [a], José Meseguer [a], Peter Csaba Ölveczky [b,*]

[a] University of Illinois at Urbana-Champaign, United States
[b] University of Oslo, Norway

HIGHLIGHTS

- Formally defines the Multirate PALS synchronizer for distributed multirate systems.
- Proves the correctness of Multirate PALS.
- Uses Multirate PALS on a distributed control algorithm for turning an airplane.

ARTICLE INFO

Article history:
Received 19 February 2013
Received in revised form 22 July 2013
Accepted 16 September 2013
Available online 4 October 2013

Keywords:
Distributed real-time systems
Multirate systems
Synchronizers
Model checking
Rewriting logic

ABSTRACT

Distributed real-time systems (DRTSs), such as avionics and automotive systems, are very hard to design and verify. Besides the difficulties of asynchrony, clock skews, and network delays, an additional source of complexity comes from the multirate nature of many such systems, which must implement several levels of hierarchical control at different rates. In previous work we showed how the design and implementation of a single-rate DRTS which should behave in a virtually synchronous way can be drastically simplified by the PALS model transformation that generates the DRTS from a much simpler synchronous model. In this work we present several simple model transformations and a multirate extension of the PALS pattern which can be combined to reduce the design and verification of a virtually synchronous multirate DRTS to the much simpler task of specifying and verifying a single synchronous system. We illustrate the ideas with a multirate hierarchical control system where a central controller orchestrates control systems in the ailerons and tail of an airplane to perform turning maneuvers.

1. Introduction

Many cyber-physical systems such as cars, airplanes, and networked medical devices are *distributed real-time systems* (DRTSs), where many components interact *asynchronously* through a network, yet must obey hard real-time synchronization constraints which are essential to their correctness. That is, they must be *virtually synchronous*. As these systems grow in complexity before our eyes, their often safety-critical nature and associated certification requirements make their development increasingly challenging, to the point where verification efforts can easily dominate the cost of system development. The accumulated complexities of concurrency, network communication, clock skews, hard real-time constraints, and synchronization constraints needed to achieve virtual synchrony make verification a daunting task, and raise the ominous specter of the "no fault found" problem: it may be extremely difficult and expensive to reproduce and locate an error, even if one is lucky enough to observe it once. To make things worse, formal verification by automatic methods such as model checking is all but impossible even for small systems, due to the state space explosion caused by asynchrony. For all these

* Corresponding author.
E-mail address: peterol@ifi.uio.no (P.C. Ölveczky).

0167-6423/$ – see front matter
http://dx.doi.org/10.1016/j.scico.2013.09.010

reasons, a component-based, modular approach to DRTS design based on highly reusable *complexity-reducing formal patterns* that can drastically reduce the effort and cost involved in DRTS design, implementation, and verification is sorely needed.

Several such formal patterns have been proposed. They offer impressive reductions in system complexity and make automatic verification possible where it was impossible before. For DRTSs that must obey virtual synchrony, both the PALS ("Physically Asynchronous, Logically Synchronous") pattern we have developed with our colleagues at UIUC and Rockwell–Collins [1,2], and the TTA pattern proposed in [3,4] can greatly reduce system complexity and make the verification of system properties much easier. For example, for a simple avionics case study considered in [2], the number of system states for their simplest possible distributed version with perfect clocks and no network delays was 3,047,832, but the PALS pattern reduced the number of states to be analyzed by model checking to a mere 185. This is certainly helpful; but the problem still remains that patterns such as PALS and TTA assume a *single period* for the virtually synchronous system. This excludes many DRTSs, in fact the majority, which do not operate at a single rate but are *multirate*. It is a fact of life that different sensors and effectors need to operate at different rates; and that this necessitates using slower rates in the distributed control hierarchies that orchestrate and synchronize their actions in, say, a car or an airplane.

The goal of the present work is to propose *Multirate PALS* as a formalized mathematical model providing a formal pattern that can drastically reduce the complexity of designing, verifying, and implementing multirate DRTSs that must achieve virtual synchrony in an asynchronous setting. In particular, we prove that the entire DRTS design as a concurrent system of asynchronous components communicating in a network is *bisimilar* to an enormously simpler *synchronous multirate ensemble* of state machines. This bisimilarity provides a very drastic reduction on the number of states, making model checking verification possible in many cases where it is unfeasible for the original DRTS. As we explain in more detail in Section 8, our work shares the same complexity-reducing goals as those of our colleagues in [5], who have made a similar, but substantially different, proposal of a multirate PALS architecture expressed in terms of the AADL modeling language. We differ from [5] not only on the model of Multirate PALS that is actually proposed, but more importantly in providing mathematical foundations for the Multirate PALS model, its asynchronous counterpart, and the bisimulation relation between both not available in [5].

Multirate PALS can drastically simplify the design, verification, and implementation of distributed cyber-physical systems whose main architecture is one of *hierarchical distributed control*. Systems of this nature are very common in avionics, motor vehicles, robotics, and automated manufacturing. Although these systems are distributed, they must achieve virtual synchrony *in real time*, since actual deadlines must be met in physical time for physical reasons. This also poses strong requirements on the network infrastructures they can use, since these must ensure message delivery and clock synchronization within precise and tight enough bounds.

Our approach, based on rewriting logic [6] and formalized in the Real-Time Maude specification language [7], is highly modular and consists of expressing Multirate PALS itself as the composition of several simple formal patterns, that is, component transformations, such as: (i) a transformation $M \mapsto M^{\times k}$ that makes a state machine k times slower; (ii) a transformation $M \mapsto M_\alpha$ that adapts the inputs of machine M according to some adaptor functions α; (iii) a transformation $\mathfrak{E} \mapsto \mathit{MRSC}(\mathfrak{E})$ that maps a *multirate ensemble* $\mathfrak{E}$ to a single state machine equivalent to its synchronous composition, where $\mathfrak{E}$ is a mathematical model of a collection of interconnected state machines running at different rates, yet synchronously in terms of their hyperperiod; and (iv) the PALS pattern itself, modified and extended to deal with faster components. Based on these modular transformations we give a formal specification of Multirate PALS as a model transformation $(\mathfrak{E}, T, \Gamma) \mapsto \mathcal{MA}(\mathfrak{E}, T, \Gamma)$, which maps a multirate ensemble $\mathfrak{E}$, plus period T and performance parameters Γ, to a semantically equivalent specification of distributed real-time components $\mathcal{MA}(\mathfrak{E}, T, \Gamma)$.

In summary, the new contributions of this work are:

1. The mathematical definitions of the patterns $M \mapsto M^{\times k}$ and $M \mapsto M_\alpha$, and of a multirate ensemble $\mathfrak{E}$ and its synchronous composition $\mathit{MRSC}(\mathfrak{E})$.
2. The mathematical definition of Multirate PALS as the transformation $(\mathfrak{E}, T, \Gamma) \mapsto \mathcal{MA}(\mathfrak{E}, T, \Gamma)$, and a *bisimulation theorem*, proving that the state machine $\mathit{MRSC}(\mathfrak{E})$ and the real-time system $\mathit{Stable}(\mathcal{MA}(\mathfrak{E}, T, \Gamma))$ associated to the stable states of $\mathcal{MA}(\mathfrak{E}, T, \Gamma)$ are bisimilar and satisfy the same *CTL** formulas.
3. An aeronautics case study showing the power of Multirate PALS in reducing the (in fact unfeasible) model checking verification of properties for the hierarchical control system involved in the turning maneuvers of an airplane to the much simpler and feasible task of model checking a single synchronous machine of the form $\mathit{MRSC}(\mathfrak{E})$.

This paper is a substantial extension of our conference paper [8]. The main new contributions are:

- More detailed specifications and mathematical proofs.
- The multirate distributed system $\mathcal{MA}(\mathfrak{E}, T, \Gamma)$ is now defined in a modular way that makes the different model transformations clear and explicit.
- The specification of the timed behavior of the distributed system was not included in [8] but is treated in detail here.
- This paper extends our previous definitions, which only covered *flat* multirate ensembles, to the much more general *hierarchical* multirate ensembles.